Credit Card Churn

Classification Analysis

Main Objective of Analysis

Using various classification models, the goal of this analysis was to determine which model provided the best results based on standard metrics to determine credit card churn. This kind of classification analysis would be useful to credit card companies to determine why customers close their accounts and assess what kinds of factors would prevent future customers from churning. The primary focus of this analysis was prediction rather than interpretation, some analysis was done on the dataset and after comparing the predictions of several different classification models, an ensemble method was utilized to combine them and assess which model overall performed the best. The model that scored the best on standard metrics for classification can later be handed over to the marketing department so they can minimize credit card churn and target potential customers with lower turnover.

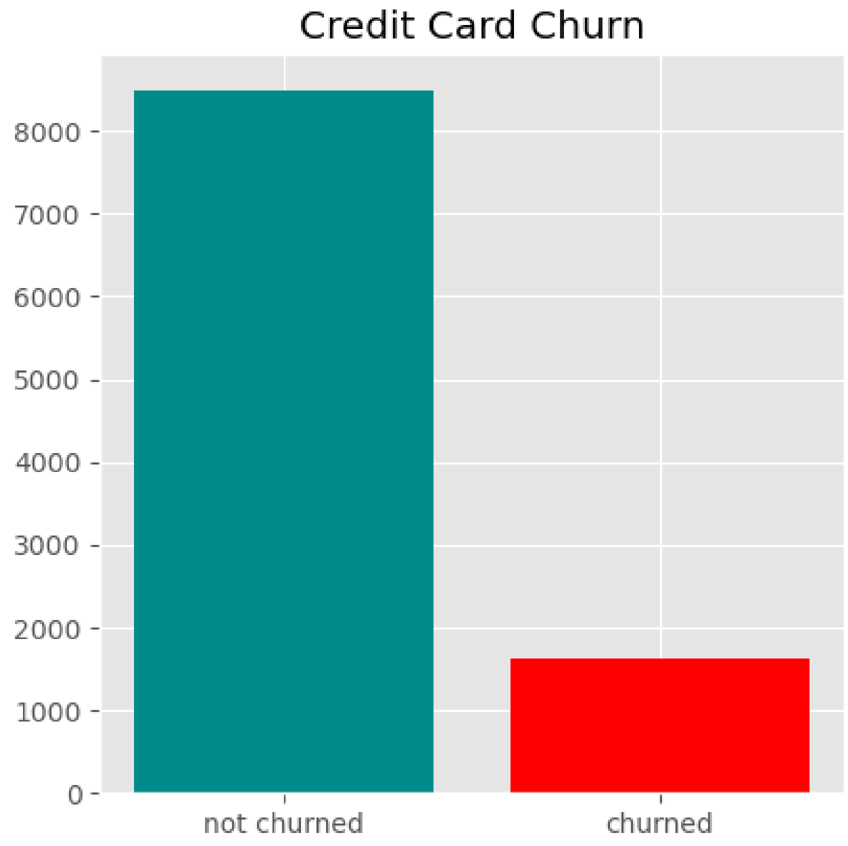
Description of Dataset

The dataset is from an anonymous source with sensitive personal information like name, birthday, and credit card number excluded. It is a sample of 10127 accounts with the following 23 attributes:

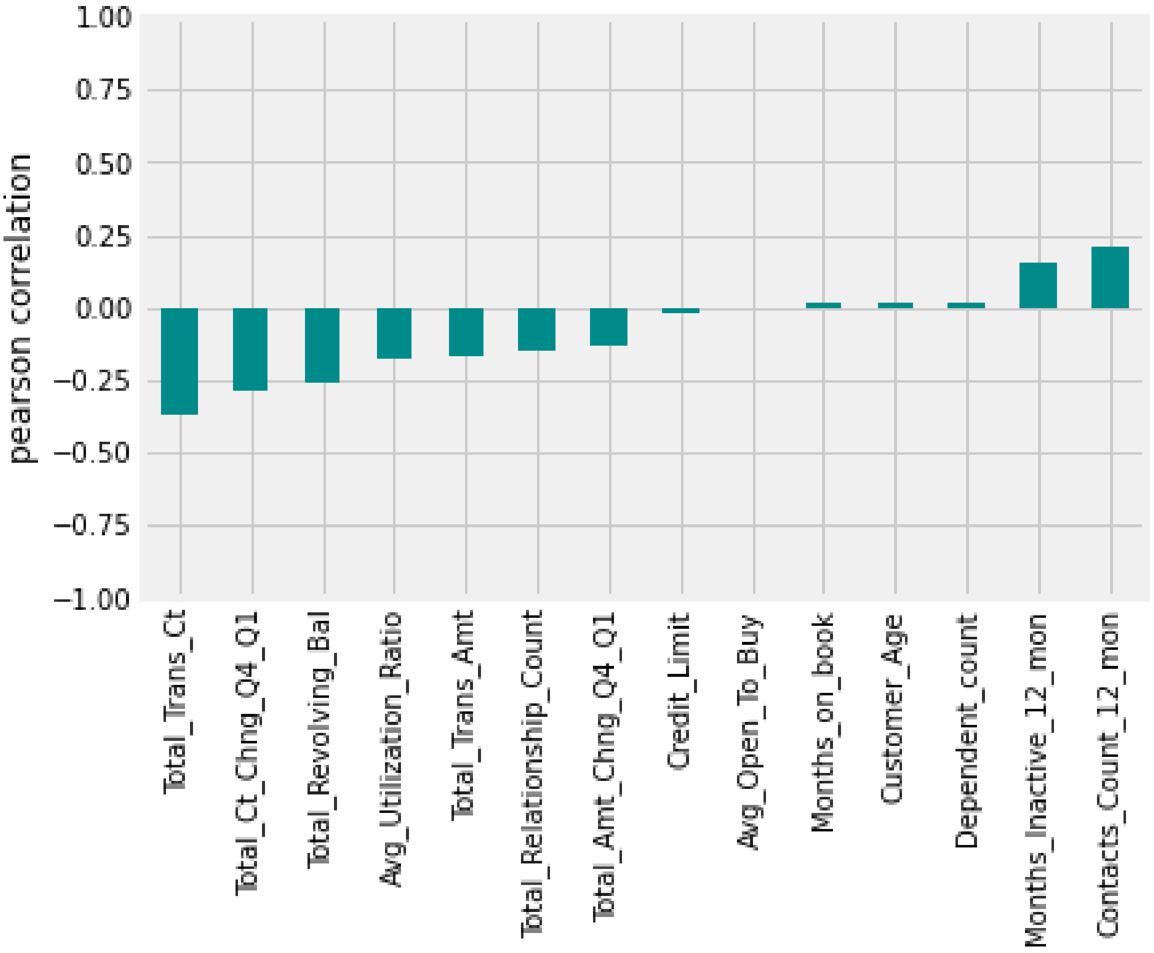
1. Client number (int64)
2. Attrition flag (object)
3. Customer age (int64)
4. Gender (object)
5. Dependent count (int64)
6. Education level (object)
7. Marital status (object)
8. Income category (object)
9. Card category (object)
10. Months on book (int64)
11. Total relationship count (int64)
12. Months inactive 12 months (int64)
13. Contacts count 12 months (int64)
14. Credit limit (float64)
15. Total revolving balance (int64)
16. Average open to buy (float64)
17. Total amount changed Q4 Q1 (float64)
18. Total transaction amount (int64)
19. Total transaction count (int64)
20. Total count changed Q4 Q1 (float64)
21. Average utilization ratio (float64)
22. Data column from previous project
23. Data column from previous project

Exploratory Data Analysis, Data Cleaning, and Feature Engineering

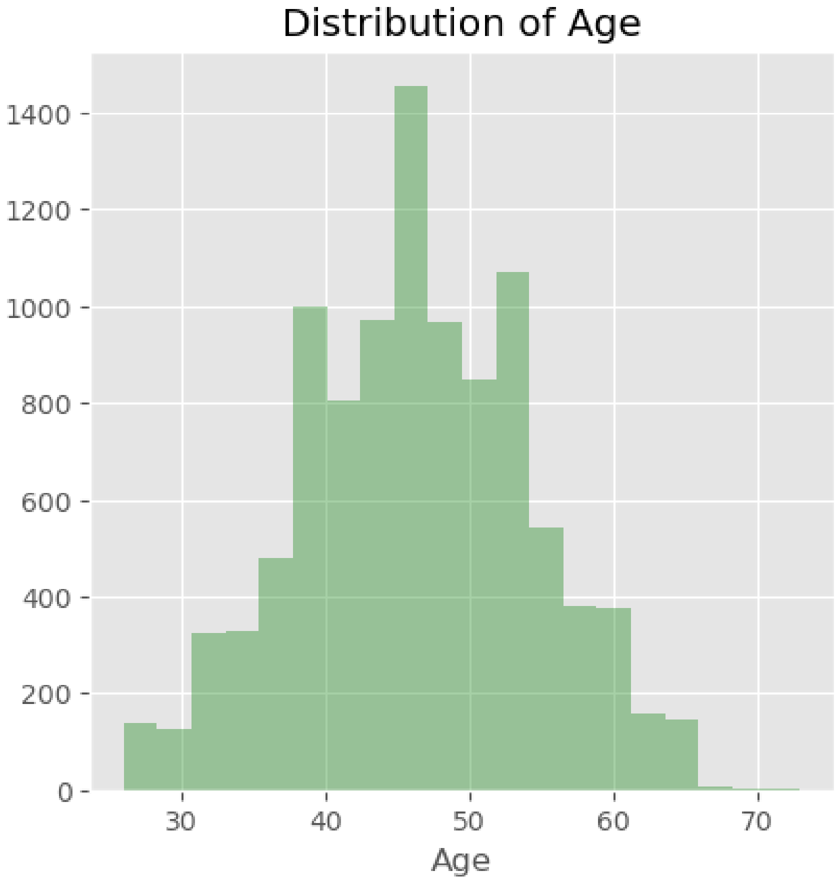
The target variable for the present analysis is the churn rate, which is indicated by Attrition flag in the dataset. Plotting the churned versus not-churned data points, the data is clearly unbalanced with not-churned data points outnumbering the churned data points and may result in some bias and variance issues, so for machine learning models, the data was balanced using SMOTE.

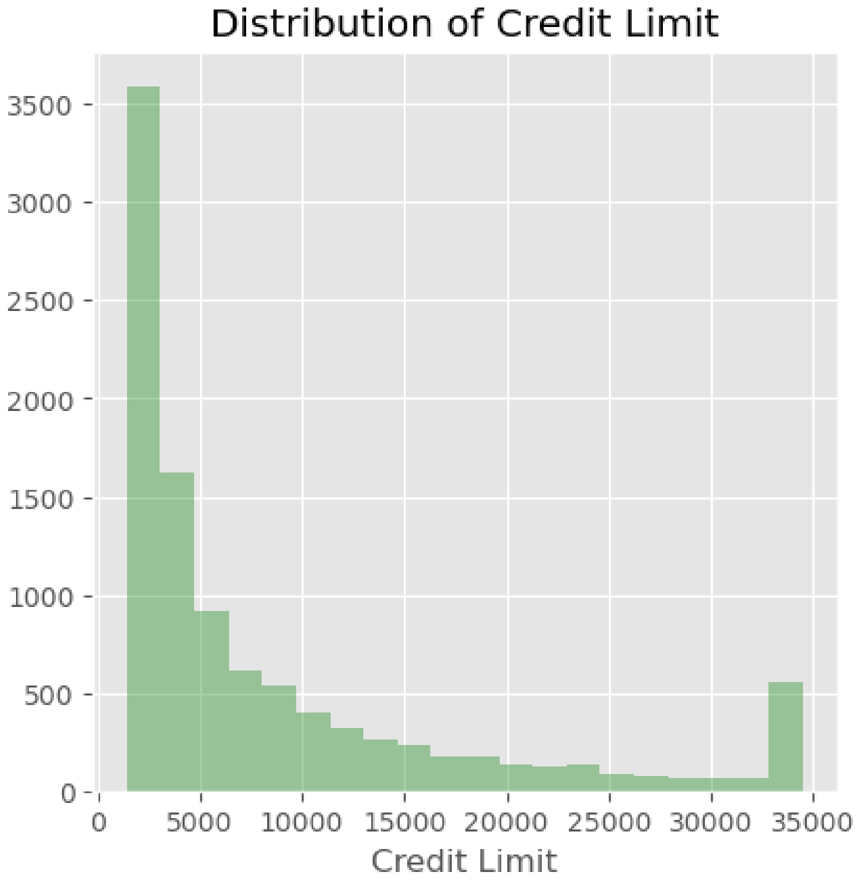


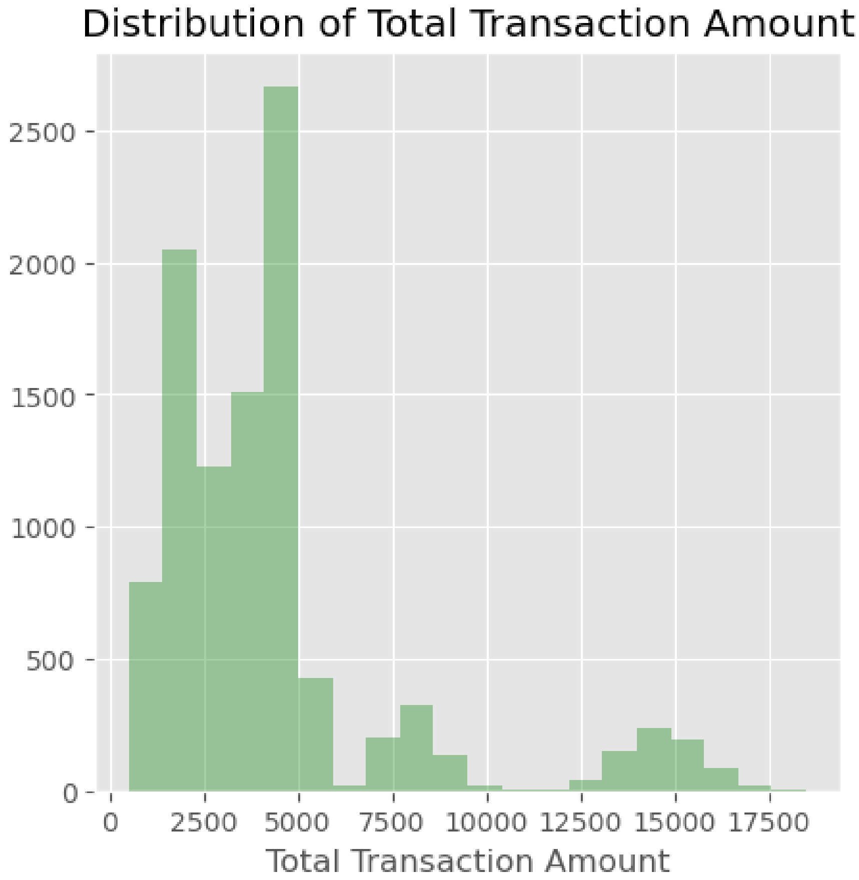
A correlation bar chart was also plotted. Higher total transaction counts reduce the tendency for a customer to churn, more so than the total transaction amount. Understandably, when an account is inactive for a year, the customer is more likely to churn.



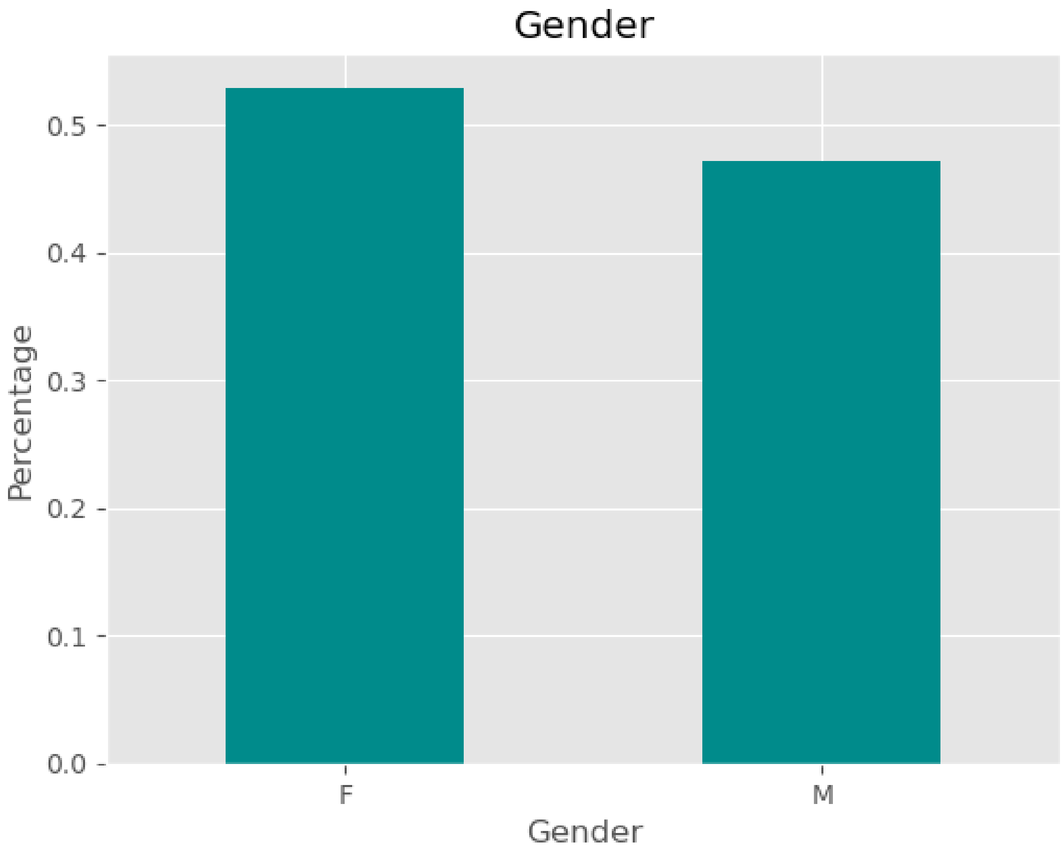
Some numerical attributes were plotted as density curves for the entire dataset and distributions varied depending on the type of attribute. With this particular credit card, customers tended to be middle aged, the credit limit was skewed, most customers had a lower credit limit, and most transaction amounts were capped around $5000.

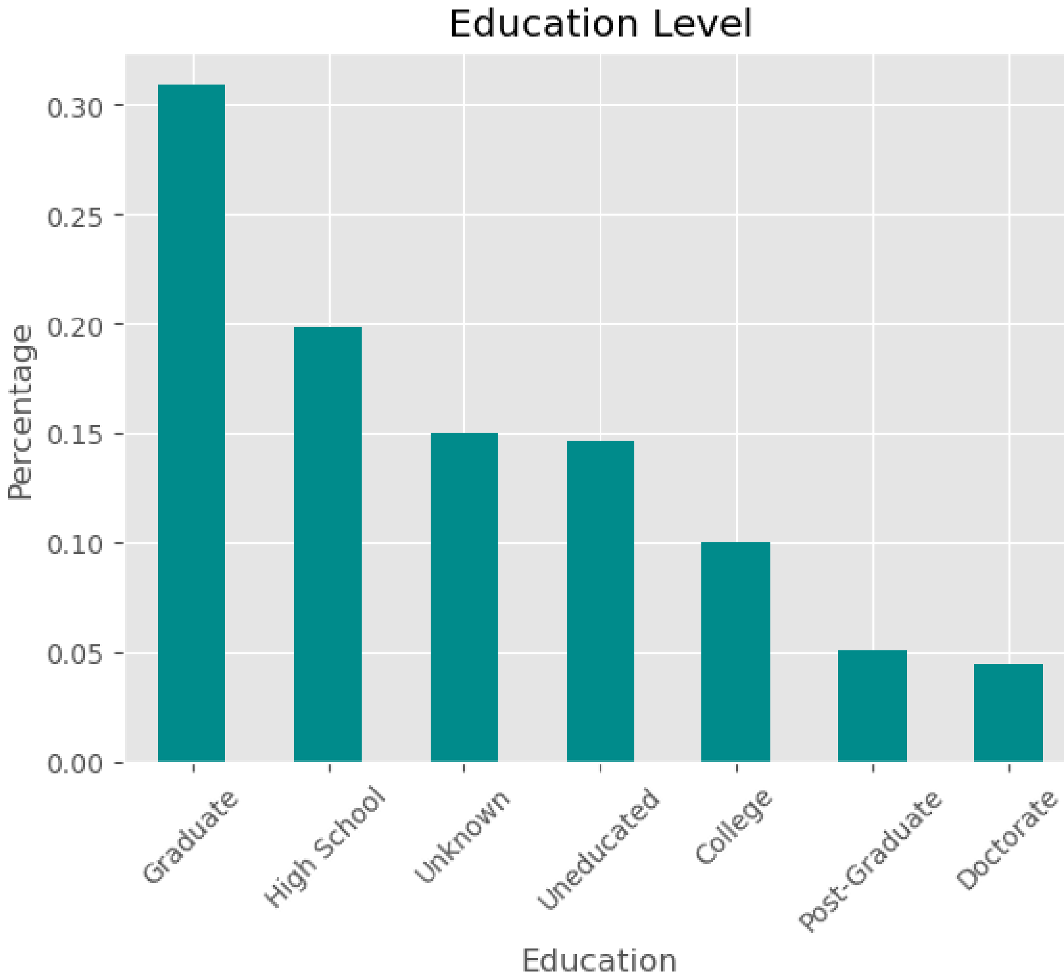


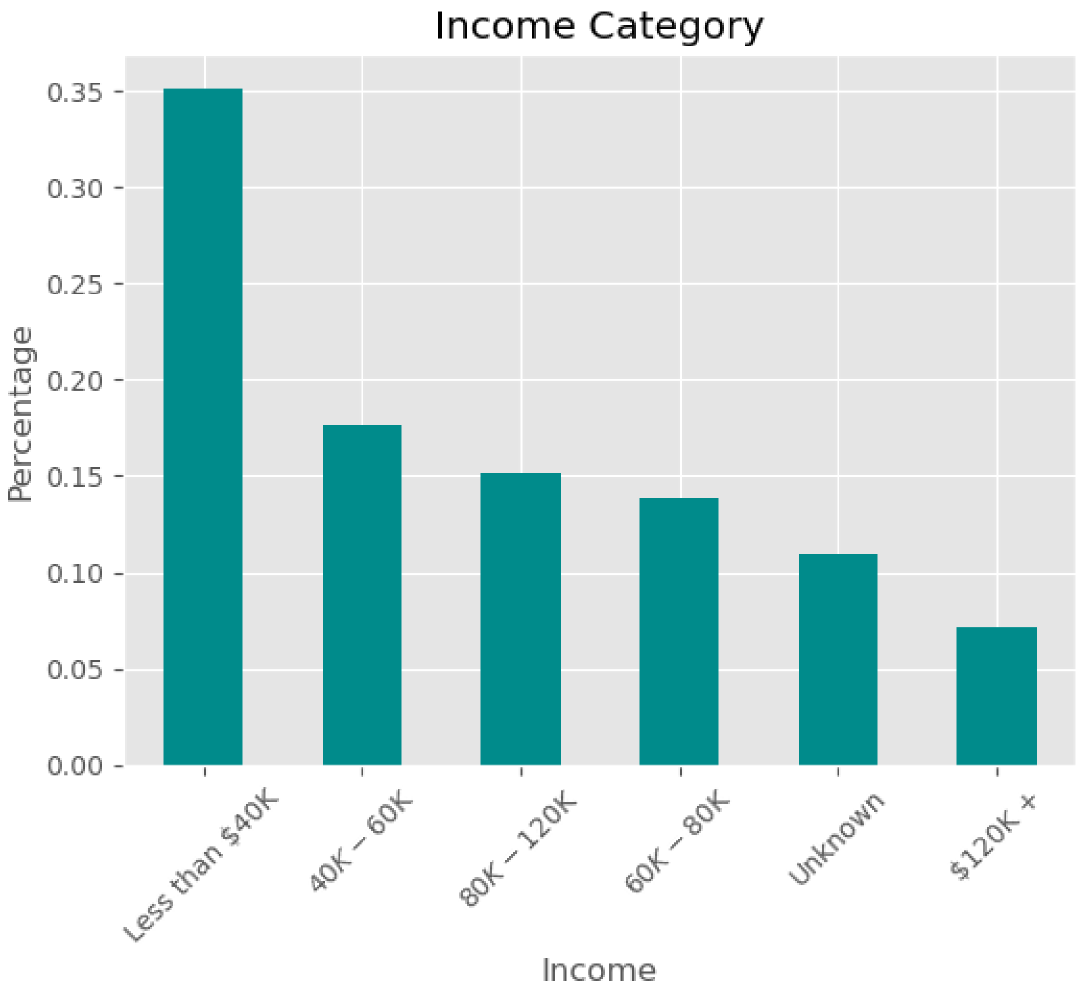

Distribution of Credit Limit
3500
3000
2500
2000
1500
1000
500
0
0
5000
10000
15000
20000
25000
30000
35000
Credit Limit

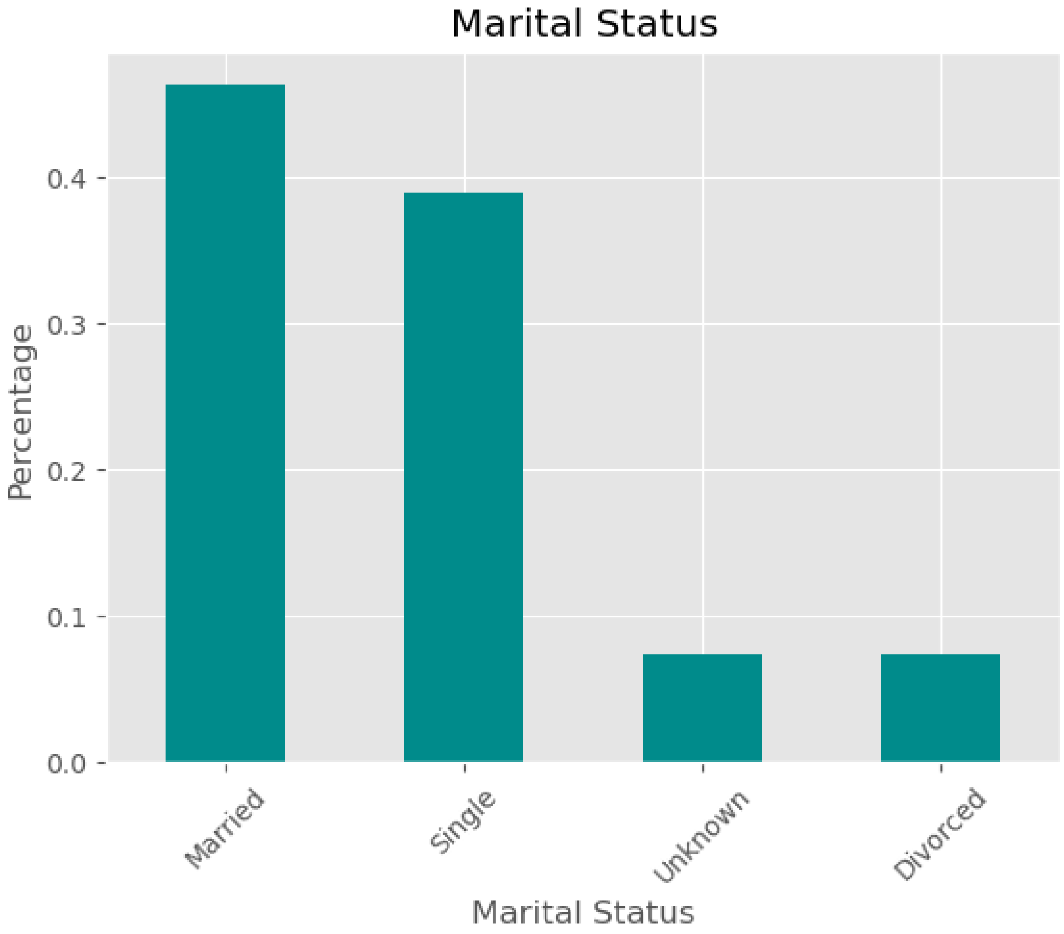

Distribution of Total Transaction Amount
2500
2000
1500
1000
500
0
0
2500
5000
7500
10000
12500
15000
17500
Total Transaction Amount

Bar charts were plotted comparing different categories within each attribute for the entire dataset. Credit card customers were close to evenly divided by gender. Interestingly, while customers with graduate education were most numerous, so were customers in the lower income bracket. Most customers were married and divorced customers were relatively few.




Education Level
Percentage
0.00
0.05
0.10
0.15
0.20
0.25
0.30
Graduate
High School
Unknown
Uneducated
College
Post-Graduate
Doctorate
Education


Income Category
Percentage
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
Less than $40K
40K − 60K
80K − 120K
60K − 80K
Unknown
$120K +
Income



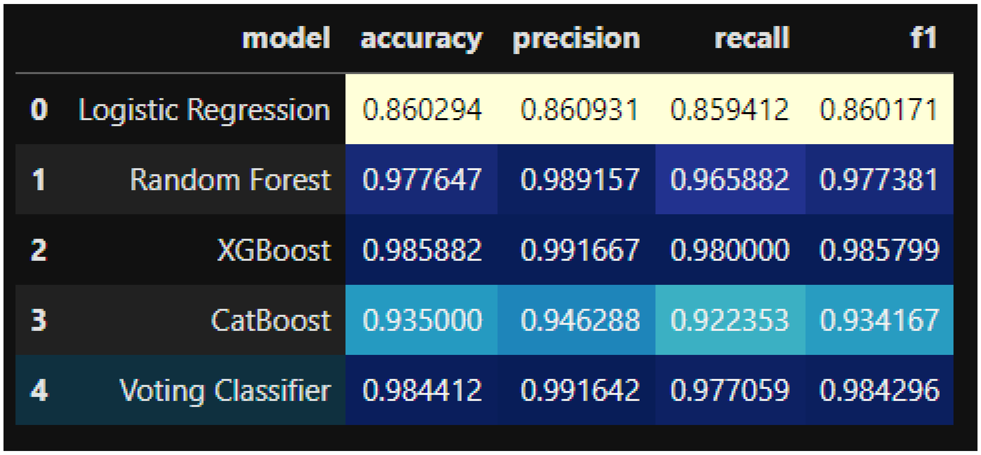
No null values were present in the dataset. Several of the categorical attributes had an 'Unknown' value which was kept and treated as its own category rather than dropping those observations entirely or replacing them with other values. The client number column was dropped as it did not provide meaningful numerical information, as well as the last two columns which were remnants of a previous project and not relevant here. For some models used in this analysis, catergorical variables must be encoded to integers, so variables with two unique values such as gender were binarized using `LabelBinarizer()` and variables with more than two unique values such as income category were encoded using `LabelEncoder()`. To split the data into train and test sets, `StratifiedShuffleSplit()` was used with one split and a test size ratio of 0.3. Finally, because variables such as age and total transaction amount were on widely different ranges of numerical values, `MinMaxScaler()` was used to scale the data so certain models used in this analysis would work better.

Classification Models

For the puposes of this analysis, the following classification models were used: logistic regression, random forest, XGBoost, CatBoost, voting classifier. Logistic regression was used as a test run and exploration of different hyperparameters was not performed.

`RandomizedSearchCV()` was used to find optimal hyperparameters for random forest, as the grid size was large and the algorighm takes more time to train. The two boosting classifiers used, XGBoost and CatBoost, `GridSearchCV()` was used to find optimal hyperparameters as they train faster than traditional tree-based algorithms like random forest. Finally, an ensemble method, voting classifier combined random forest and XGBoost, the two models that performed the best individually.

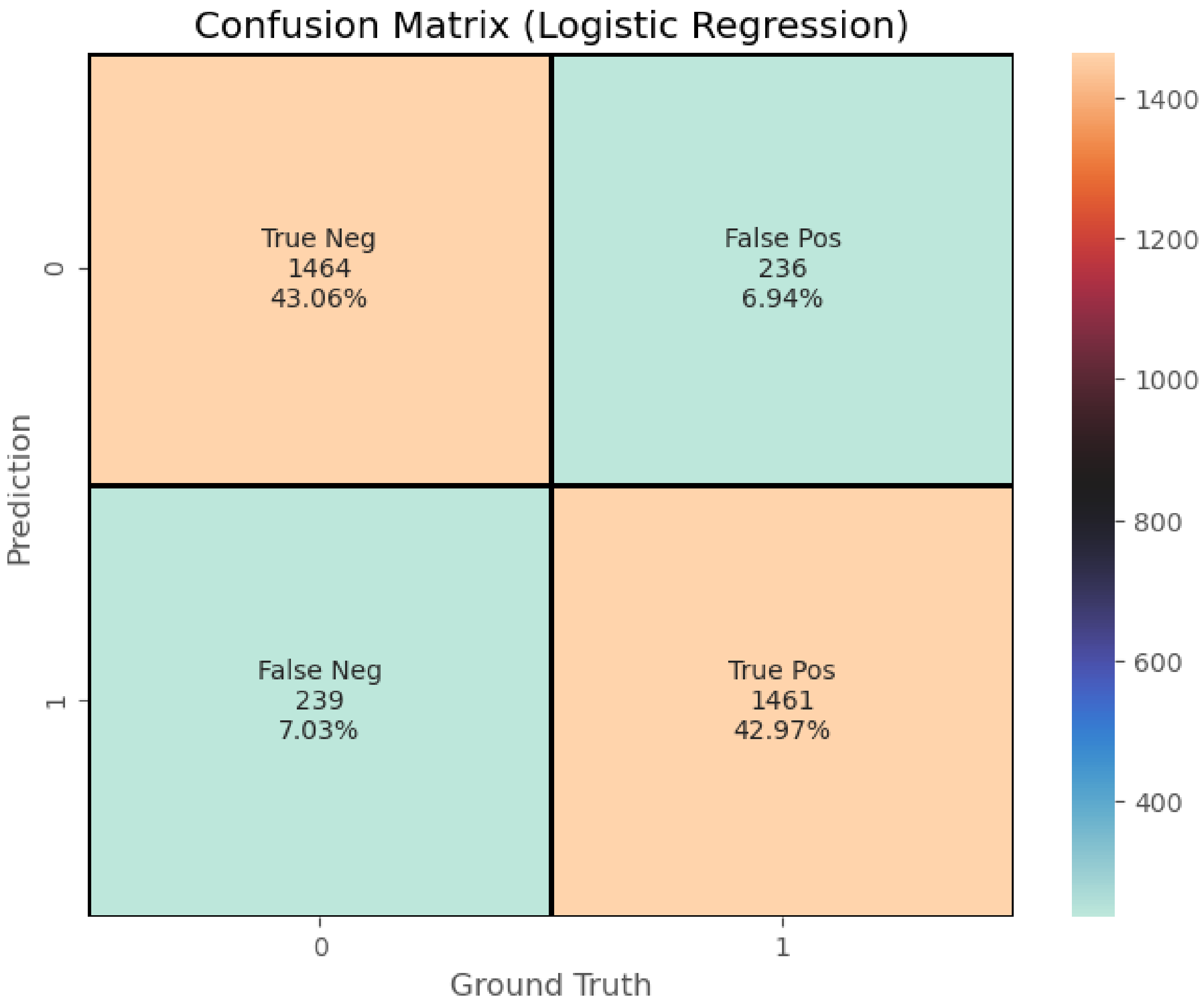
The following are the performance metrics of the trained models:

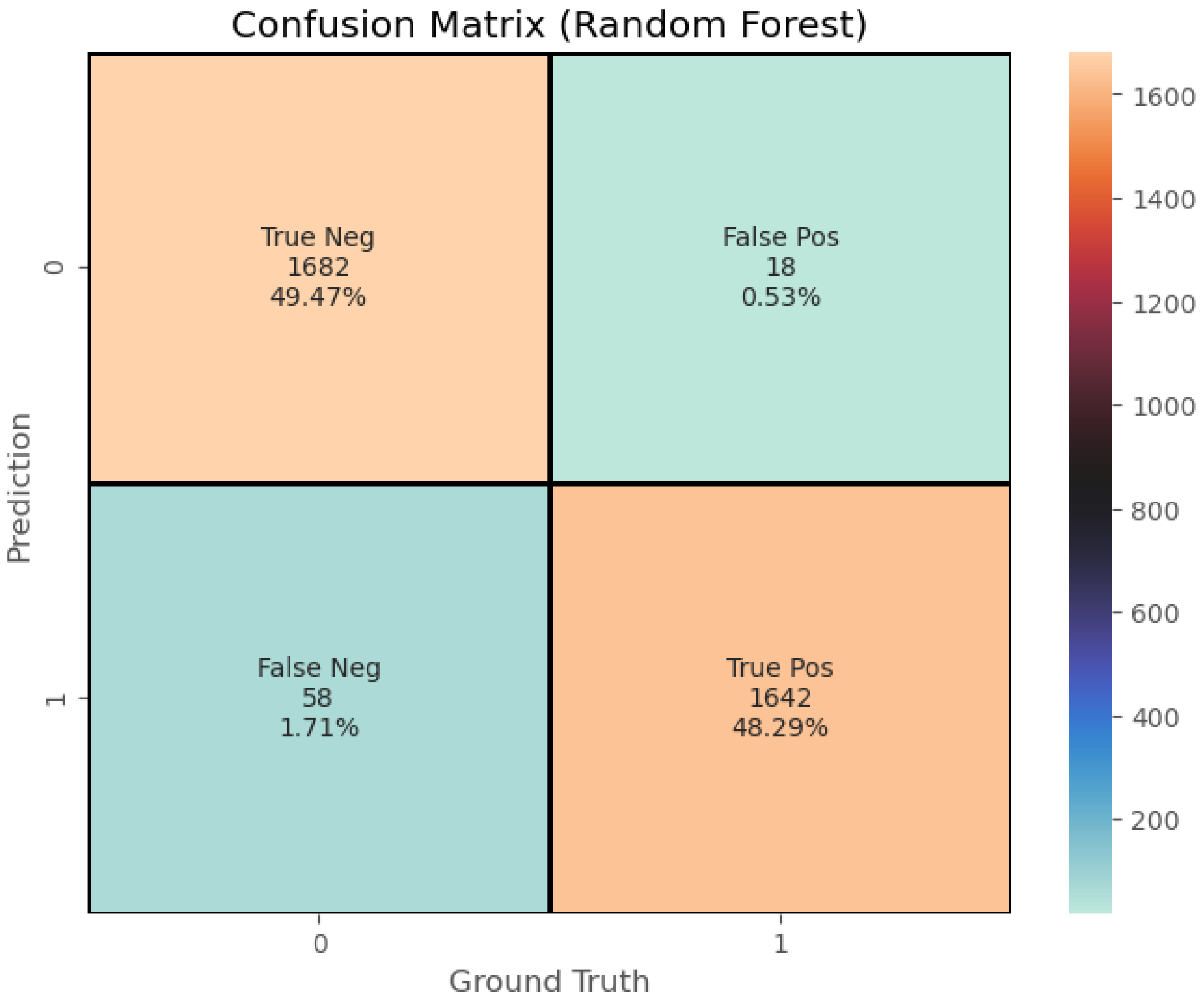


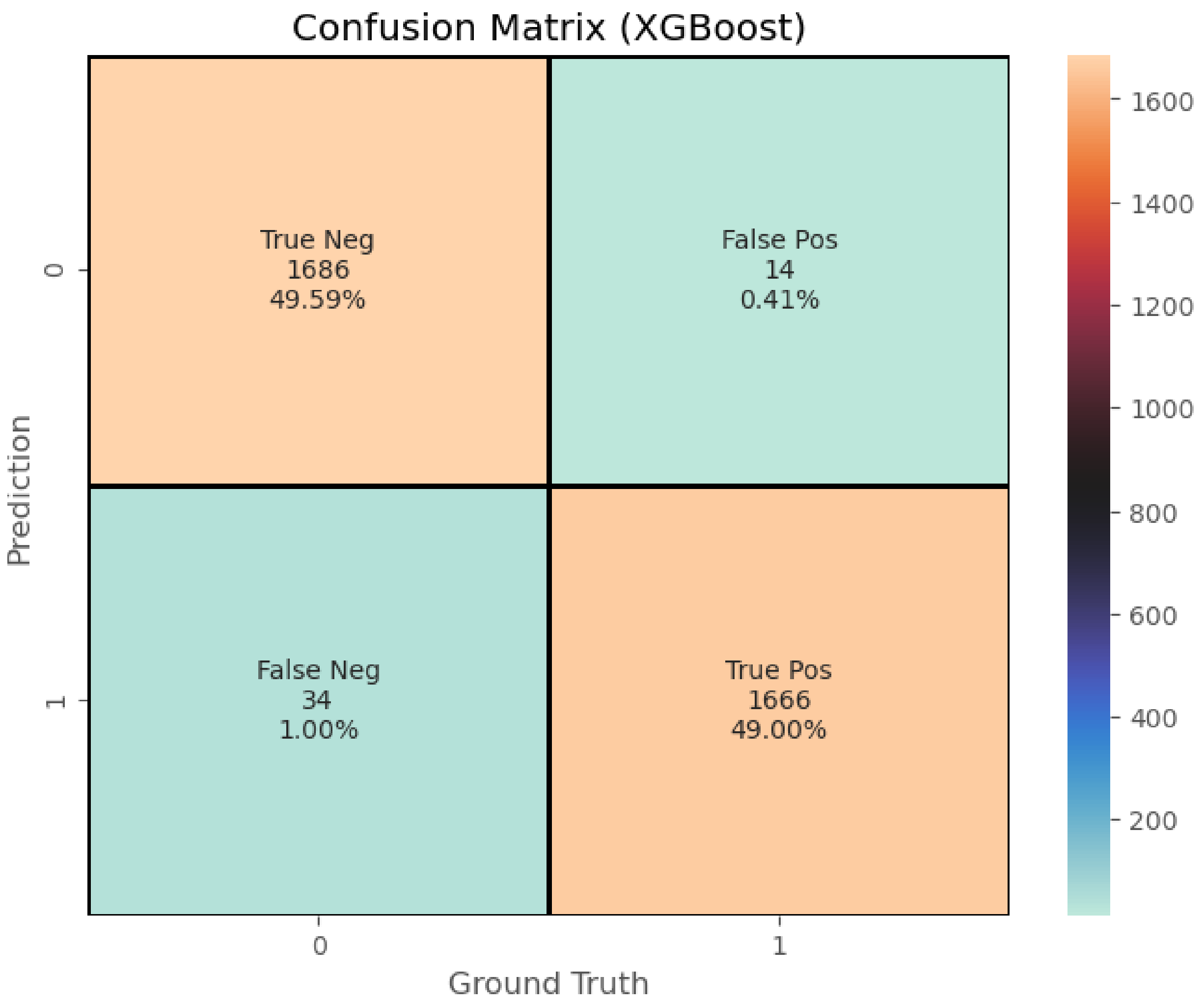
	model	accuracy	precision	recall	f1
0	Logistic Regression	0.860294	0.860931	0.859412	0.860171
1	Random Forest	0.977647	0.989157	0.965882	0.977381
2	XGBoost	0.985882	0.991667	0.980000	0.985799
3	CatBoost	0.935000	0.946288	0.922353	0.934167
4	Voting Classifier	0.984412	0.991642	0.977059	0.984296

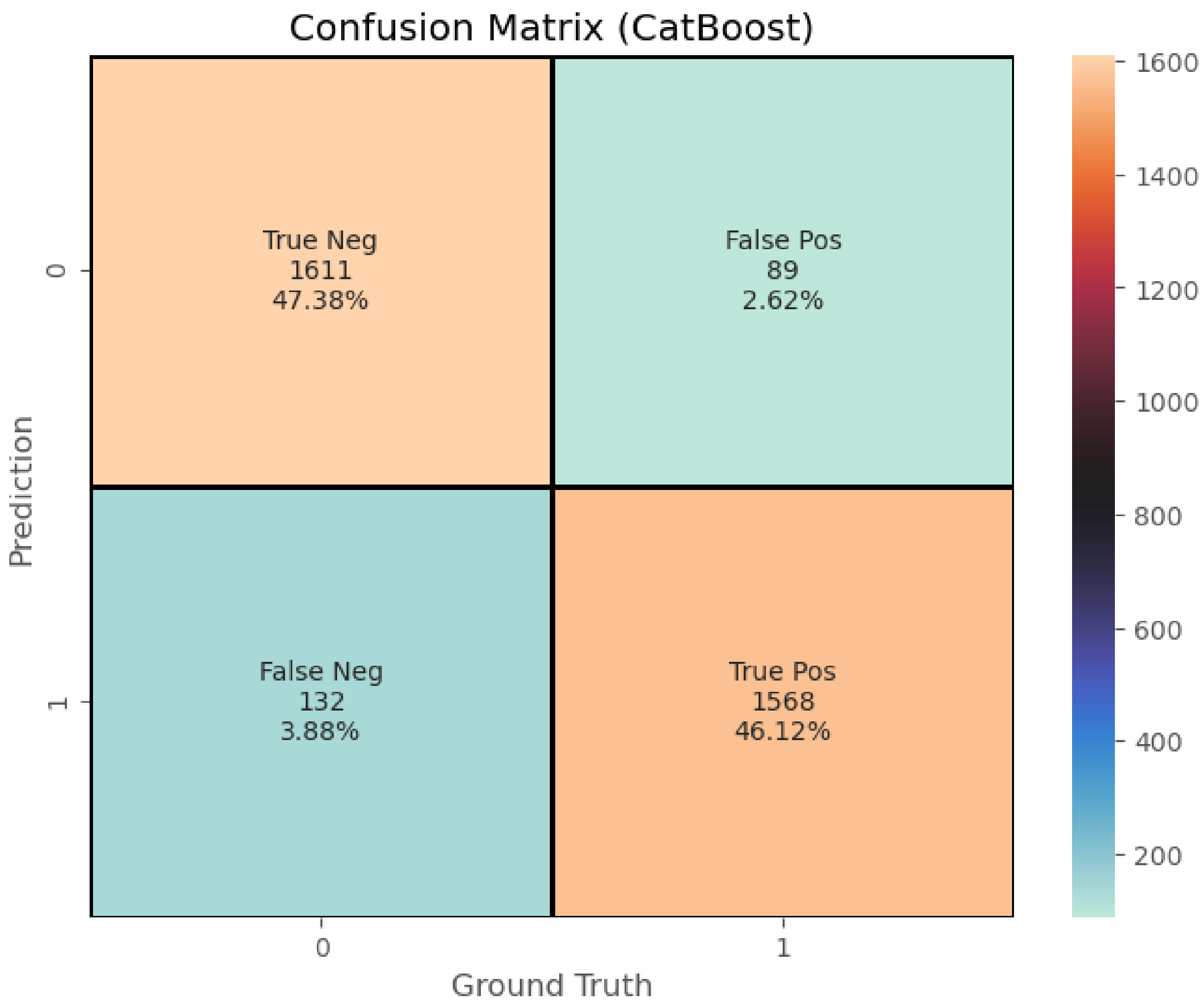
Overall, XGBoost performed the best with highest accuracy, precision, recall, and F1 scores. The voting classifier combing random forest and XGBoost came a close second. The motivation behind using an ensemble of both random forest and XGBoost was to increase scores, but the actual result was that random forest interferred with XGBoost rather than compliment it.

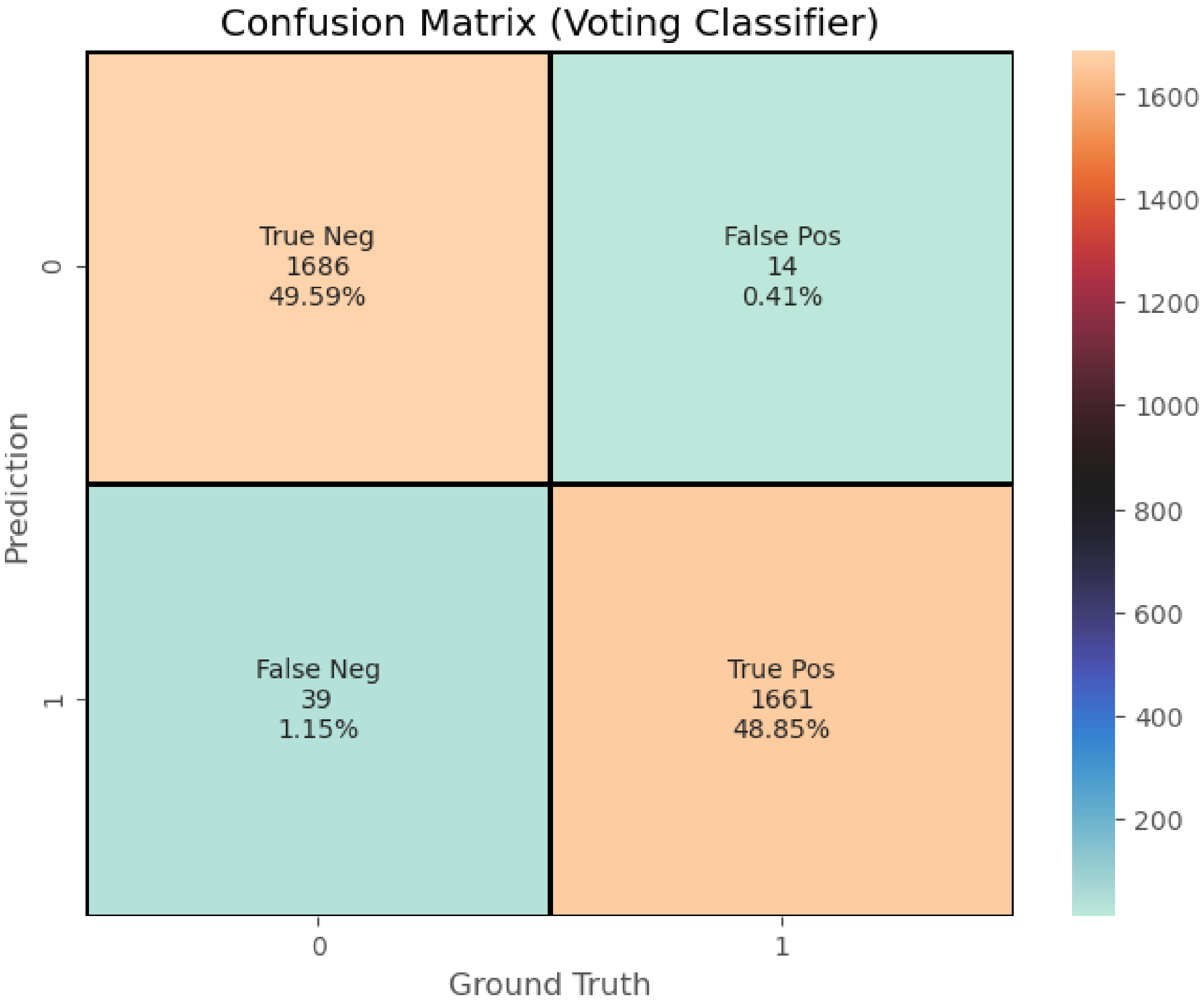
The following are confusion matrices for the different models, comparing true positive, true negative, false positive, false negative.



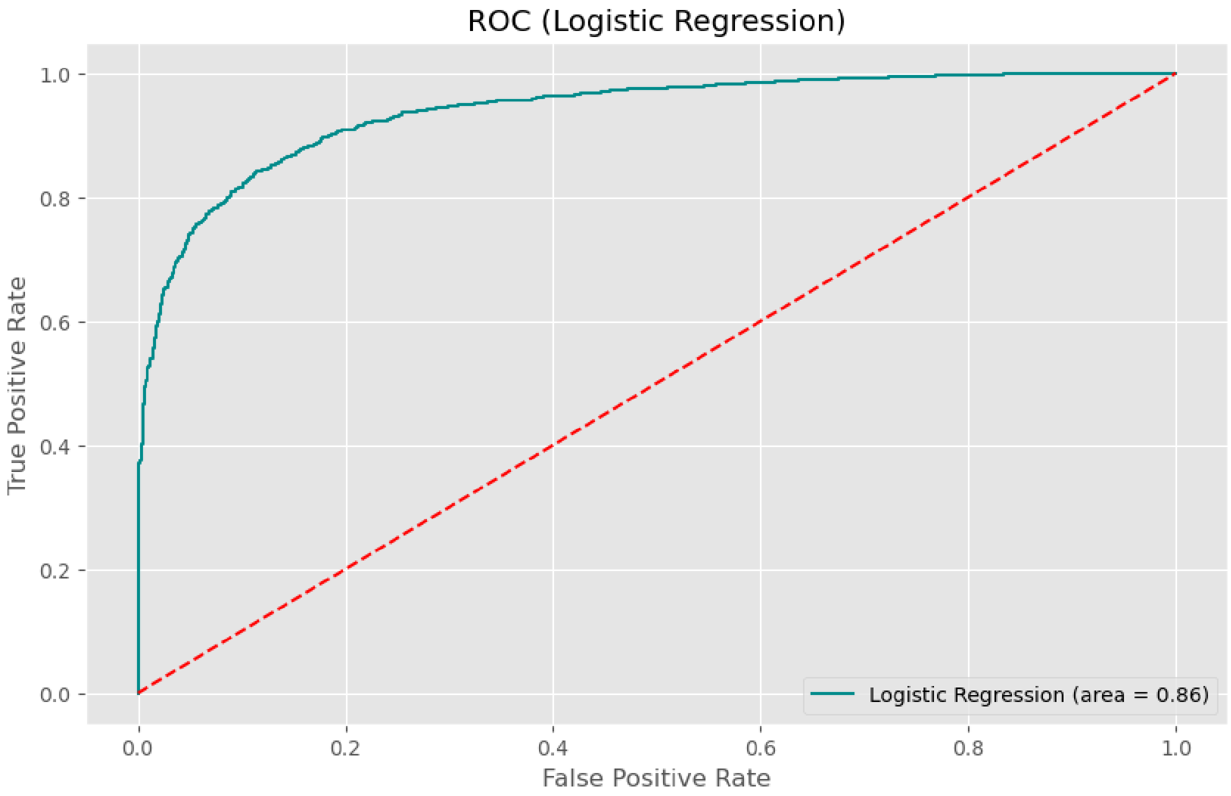

Confusion Matrix (Random Forest)
True Neg
1682
49.47%
False Pos
18
0.53%
False Neg
58
1.71%
True Pos
1642
48.29%
Prediction
0
1
Ground Truth
0
1
1600
1400
1200
1000
800
600
400
200

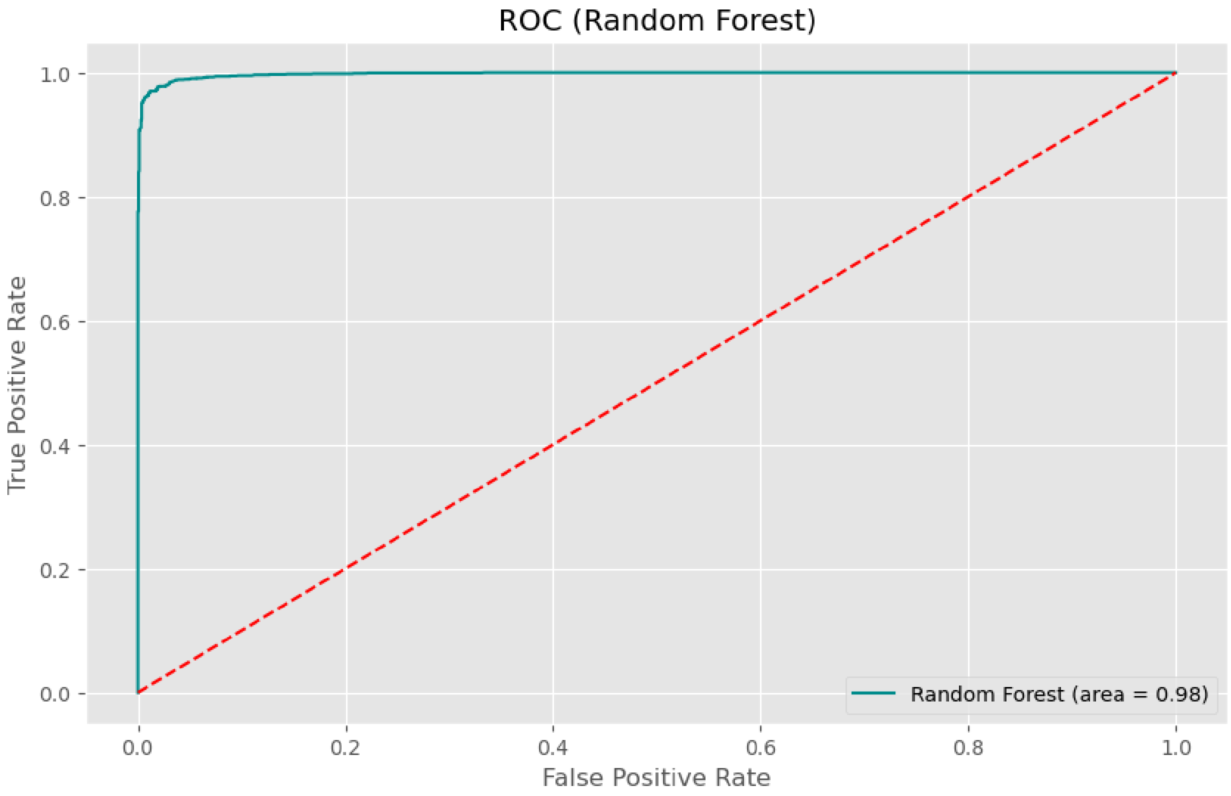

Confusion Matrix (XGBoost)
True Neg
1686
49.59%
False Pos
14
0.41%
False Neg
34
1.00%
True Pos
1666
49.00%
Prediction
0
1
Ground Truth
0
1
1600
1400
1200
1000
800
600
400
200

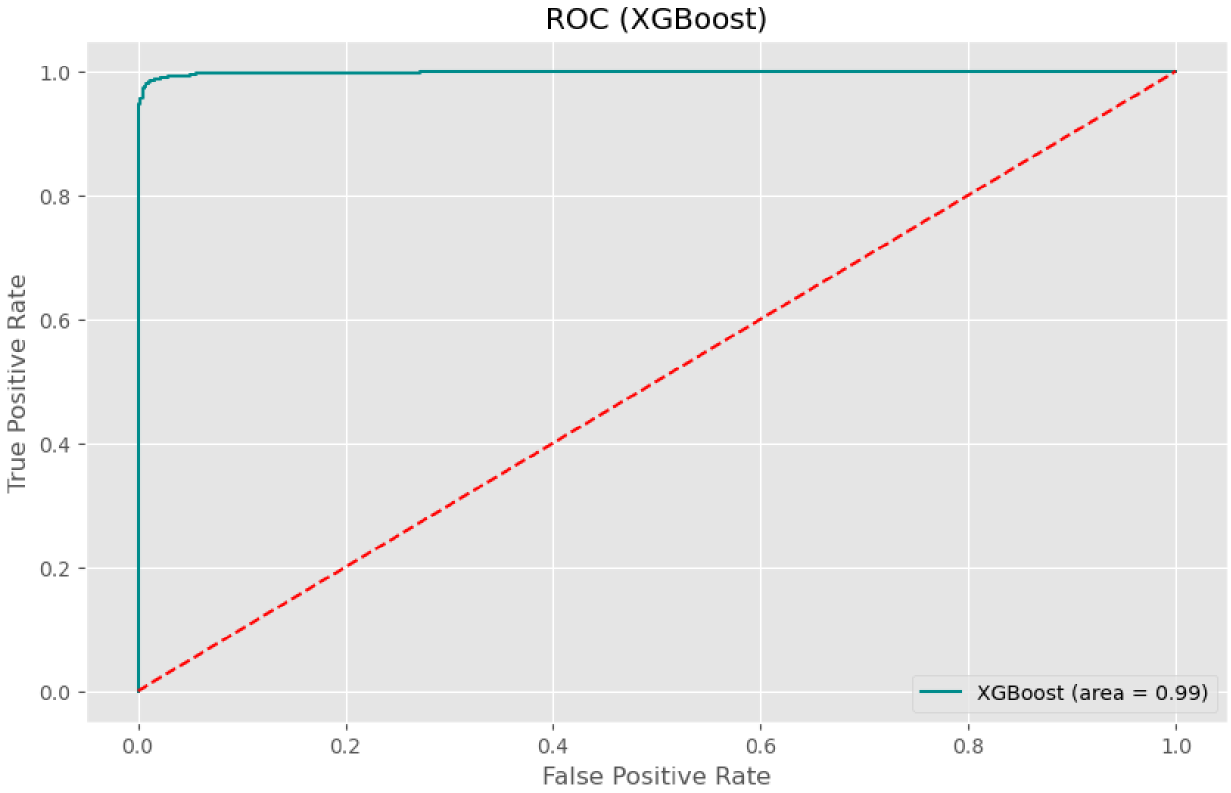

Confusion Matrix (CatBoost)
True Neg
1611
47.38%
False Pos
89
2.62%
False Neg
132
3.88%
True Pos
1568
46.12%
Prediction
0
1
Ground Truth
0
1
1600
1400
1200
1000
800
600
400
200

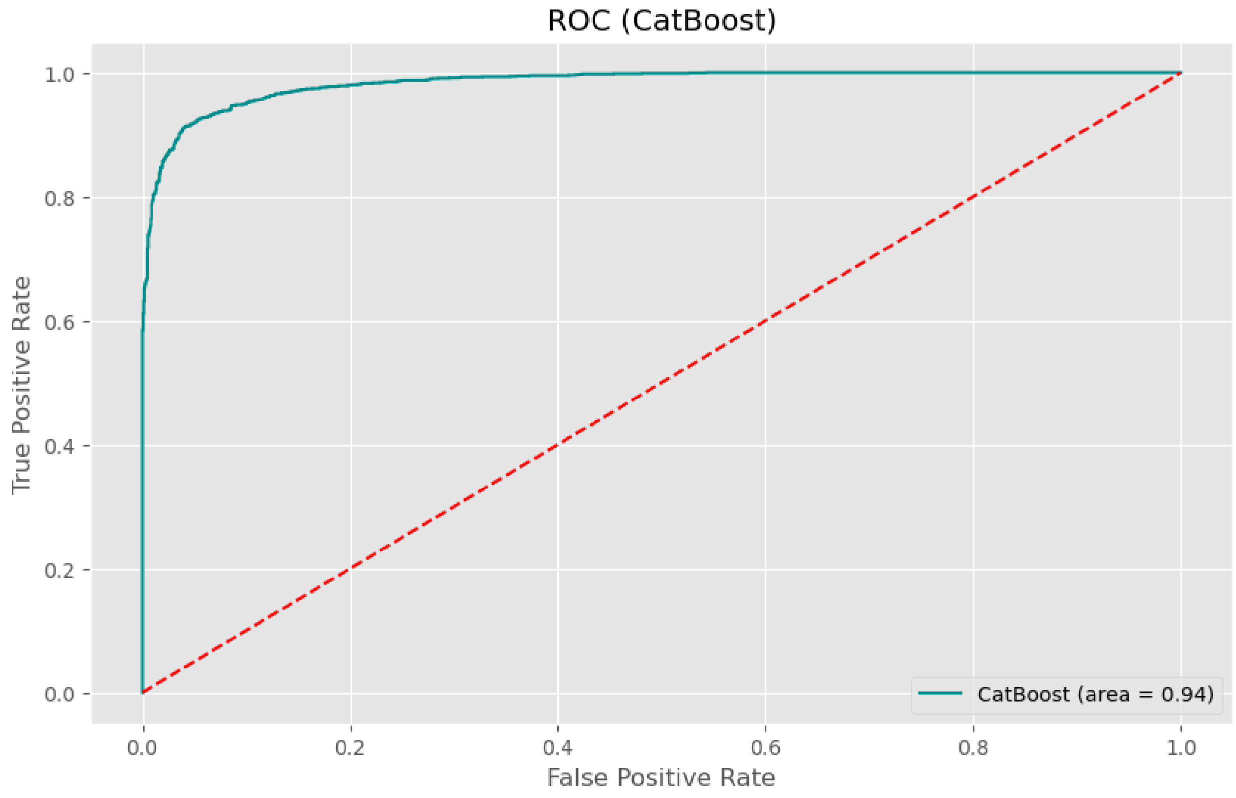

Confusion Matrix (Voting Classifier)
True Neg
1686
49.59%
False Pos
14
0.41%
False Neg
39
1.15%
True Pos
1661
48.85%
Prediction
0
1
Ground Truth
0
1
1600
1400
1200
1000
800
600
400
200

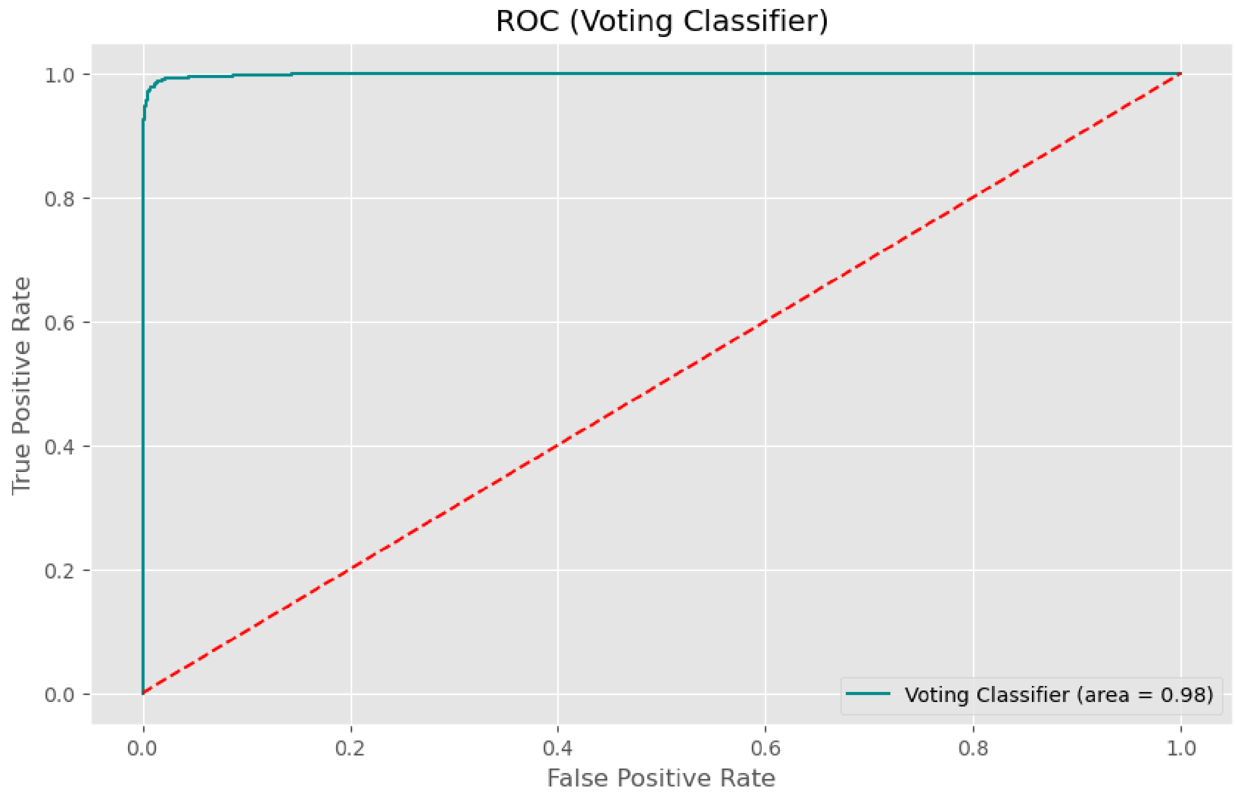
The following are the ROC curves for the different models. Because the dataset was balanced using SMOTE, an ROC curve was used rather than a precision-recall curve. As the metrics indicate, in the worst performing model, logistic regression, the curve veers furthest from true positive rate and closer to the false positive rate. And conversely, in the best performing models, XGBoost and voting classifier, the curves hug most tightly to the upper left corner.




ROC (Random Forest)
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0
Random Forest (area = 0.98)


ROC (XGBoost)
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0
XGBoost (area = 0.99)


ROC (CatBoost)
True Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
CatBoost (area = 0.94)


ROC (Voting Classifier)
True Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
Voting Classifier (area = 0.98)

Key Findings and Insights

Taking into consideration the accuracy, precision, recall, and F1 scores as well as the visualizations for the confusion matrices and the ROC curves, the various classification metrics differ and the purely numeric models such as logistic regression performed more poorly than a meta-classifier like random forest or boosting classifiers like XGBoost and CatBoost which don't require feature encoding or scaling. The fact that the random forest classifier de-correlates trees and creates a random subset of features for each tree, thus reducing errors may have played a key role in making it a better classification model, as is the case with tree-based boosting classifiers like XGBoost and CatBoost in this analysis of credit card churn, which included a fair amount of categorical variables. Interestingly, the voting classifier which incorporated the XGBoost model as part of its ensemble didn't perform as well as XGBoost alone, indicating that the random forest model in the voting classifier served to be confounding and shows that a more complex model isn't necessarily a better one.

Next Steps

There were issues in this analysis that were not addressed. Many of the categorical variables in the dataset contained "Unknown" as values, which could mean a lot of different things. The dataset could be cleaned further in preprocessing to either replace "Unknown" with another value, or drop observations completely, although that might result in a reduction of the dataset significant enough to decrease the performance of classification models. Additional data could be collected and used for the analysis. Also, in exploratory data analysis, rather than plot graphs of the entire dataset, target the subset of churned customers in order to make further observations. Deep learning methods are popular lately, but if more algorithms were to be trained, to avoid deep learning models unless they are used as an ensemble with traditional machine learning models, as deep learning models alone do not perform as well on tabular data compared to traditional machine learning models.